AFA

PATENT
Attorney Docket No.: SONY-12100

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Group Art Unit: 2163
Scott D. Smyers et al.	Examiner: Filipczyk. Marcin R.
Serial No.: 09/608,617	
Filed: June 30, 2000	**TRANSMITTAL LETTER**
For: **METHOD OF AND APPARATUS FOR WRITING AND READING TIME SENSITIVE DATA WITHIN A STORAGE DEVICE**	162 N. Wolfe Road Sunnyvale, CA 94086 (408) 530-9700
	Customer No. 28960

MS: Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Enclosed please find an appeal brief in response to the notice of appeal filed June 12, 2006 for filing in the U.S. Patent and Trademark Office. Also enclosed is a check in the amount of $500.00 to cover the appeal brief filing fee.

The Commissioner is hereby authorized to charge any additional fee or credit overpayment to our Deposit Account No. 08-1275. **An originally executed duplicate of this transmittal is enclosed for this purpose.**

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: July 24, 2006 By: Jonathan O. Owens
Jonathan O. Owens
Reg No.: 37,902

Attorneys for Applicants

CERTIFICATE OF MAILING (37 CFR§ 1.8(a))
I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the U.S. Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450

HAVERSTOCK & OWENS LLP.
Date: 7/24/06 By:

PATENT
Attorney Docket No.: SONY-12100

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Group Art Unit: 2163
Scott D. Smyers et al.	Examiner: Filipczyk, Marcin R.
Serial No. 09/608,617	**APPEAL BRIEF**
Filed: June 30, 2000	
For: **METHOD OF AND APPARATUS FOR WRITING AND READING TIME SENSITIVE DATA WITHIN A STORAGE DEVICE**	162 North Wolfe Road Sunnyvale, California 94086 (408) 530-9700
	Customer No.: 28960

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In furtherance of the Applicants' Notice of Appeal filed on June 12, 2006, this Appeal Brief is submitted. This Appeal Brief is submitted in support of the Applicants' Notice of Appeal, and further pursuant to the final rejection mailed on April 14, 2006, in which Claims 1-15, 19-35 and 44-54 were rejected. The Applicants submit this Appeal Brief to the Board of Patent Appeals and Interferences in compliance with the requirements of 37 C.F.R. § 41.37, as stated in *Rules of Practice Before the Board of Patent Appeals and Interferences (Final Rule)*, 69 Fed. Reg. 49959 (August 12, 2004). The Applicants contend that the rejections of Claims 1-15, 19-35 and 44-54 in this proceeding are in error and are overcome by this appeal.

07/28/2006 BABRAHA1 00000022 09608617
01 FC:1402 500.00 OP

CERTIFICATE OF MAILING (37 CFR§ 1.8(a))
I hereby certify that this paper (along with any referred to as being attached or enclosed) is being deposited with the U.S. Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450

HAVERSTOCK & OWENS LLP

Date: 7/24/06 By:

I. REAL PARTIES IN INTEREST

As the assignee of the entire right, title, and interest in the above-captioned patent application, the real parties in interest in this appeal, is:

Sony Corporation, a Japanese corporation
6-7-35 Kitashinagawa, Shinagawa
Tokyo, 141
Japan

Sony Electronics Inc., a corporation of the State of Delaware
1 Sony Drive
Park Ridge, NJ 07656-8003

per the assignment document filed on June 30, 2000.

II. RELATED APPEALS AND INTERFERENCES

The Applicants are not aware of any other appeals or interferences related to the present application.

III. STATUS OF THE CLAIMS

Claims 1-15, 19-35 and 44-54 are pending in this case. Claims 1, 24, 30, 44 and 50-54 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,438,604 to Kuver et al. ("Kuver", a copy of which is attached as Exhibit A). Claims 1-13, 19-26, 29-32, 35 and 44-54 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Applicant Admitted Prior Art (AAPA) in view of U.S. Patent No. 6,012,117 to Traw et al. ("Traw", a copy of which is attached as Exhibit B). Claims 14, 15, 27, 28, 33 and 34 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over AAPA in view of Traw and further in view of Kuver. Within this Appeal Brief, the rejection of Claims 1-15, 19-35 and 44-54 is appealed.

IV. STATUS OF THE AMENDMENTS FILED AFTER FINAL REJECTION

No amendments have been filed after the Office Action mailed on April 14, 2006.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The invention disclosed in the present application number 09/608,617 is directed to A media storage device that records a meta-data header with packets received by the media storage device. The meta-data headers include a cycle mark value and a cycle count value. The cycle mark value has a specific pattern which is then used to locate cycle boundaries within the recorded stream of data. The cycle count value specifies the value of the isochronous cycle number on which the packet was received. The media storage device includes an embedded stream processor which is responsible for appropriately adding the meta-data header to the packets within the recorded stream of data. The embedded stream processor is also integral to the playback of recorded data, and is used to retrieve data from the storage media, strip the meta-data headers from retrieved data being played back and recover from any error conditions encountered during the playback of previously recorded data. The meta-data headers stored within the recorded stream of data are also utilized to recover from any error conditions and resynchronize the transmission of the data during playback.

The elements of Claim 1, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The method of writing data to a media storage device described there comprises receiving a received packet (74) of data to be written to the media storage device (40), the received packet (74) of data including a packet header (76), adding a meta data header (82) to the received packet (74) of data thereby forming an extended packet (80) of data including both the packet header (76) and the meta data header (82) and storing the extended packet (80) of data onto a media (48) within the media storage device (40).

The elements of Claim 8, directed to one embodiment of the present invention, are described in the Specification at page 13, line 27 through page 14, line 13; page 15, line 11 through page 16, line 16; page 17, line 14 through page 18, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The method of reading data from a media storage device which has previously been stored with header data generated by the media storage device described there comprises locating a first header data (82), including a cycle mark value having a pattern, reading a previously stored packet (80) of data following the first header data (82) from a media

(48) within the media storage device (40), the previously stored packet (80) of data including a packet header (76), stripping the first header data (82) from the previously stored packet (80) of data thereby forming a retrieved packet (74) of data and transmitting the retrieved packet (74) of data to another device.

The elements of Claim 19, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The meta data header (82) added to received packets (80) by a media storage device (40) as the packets (80) are recorded on storage media (48) within the media storage device (40), each of the received packets (80) including an existing header (76) to which the meta data header (82) is added such that the received packets (80) include both an existing header (76) and a meta data header (82), the meta data header (82) described there comprises a cycle mark value including a pattern used to locate cycle boundaries within the received packets (80) and a cycle count value specifying a cycle number of a cycle in which the received packets (80) are received.

The elements of Claim 24, directed to one embodiment of the present invention, are described in the Specification at page 13, line 1 through page 15, line 10; page 16, line 17 through page 17, line 7; page 17, line 14 through page 18, line 14; page 20, line 24 through page 21, line 13; page 22, lines 1-19 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The media storage device described there comprises means for interfacing (42) configured for receiving a stream of data, thereby forming a received stream of data, and also for transmitting a retrieved stream of data, the received stream of data including packet header data (76); means for storing (48) data for storing and retrieving the received stream of data; and means for processing (44) coupled to the means for interfacing (42) and to the means for storing (48) for adding meta header data (82) to the received stream of data as the received stream of data is received, such that each packet (80) within the received stream of data includes both packet header data (76) and meta header data (82), and providing the meta header data (82) and the received stream of data to the means for storing (48) for recording thereby forming a recorded stream of data, the meta header data (82) including a cycle mark value marking cycle boundaries within the recorded stream of data.

Means for interfacing referred to in the specification as an interface circuit (42) is shown in Figure 3. The IEEE 1394-1995 serial bus interface circuit 42 receives a stream of data from the IEEE 1394-1995 serial bus. The interface circuit 42 then forwards this data to the embedded stream processor 44. [Present Specification, page 13, lines 21-23]

Means for storing referred to in the specification as a storage media (48) is shown in Figure 3. The storage management circuit 46 then manages the storage of the stream of data including the meta-data headers onto the storage media 48. [Present Specification, page 13, lines 25-26]

Means for processing referred to in the specification as an embedded stream processor (44) is shown in Figure 3. The embedded stream processor 44 adds the meta-data headers into the stream of data, as appropriate. [Present Specification, page 13, lines 23-25]

The elements of Claim 30, directed to one embodiment of the present invention, are described in the Specification at page 13, line 1 through page 15, line 10; page 16, line 17 through page 17, line 7; page 17, line 14 through page 18, line 14; page 20, line 24 through page 21, line 13; page 22, lines 1-19 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The media storage device described there comprises an interface circuit (42) configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data (76), storage media (48) configured to store and retrieve the received stream of data and an embedded stream processor (44) coupled to the interface circuit (42) and to the storage media (48) to add meta header data (82) to the received stream of data as it is received, such that each packet within the received stream of data includes both packet header data (76) and meta header data (82), and provide the meta header data (82) and the received stream of data to the storage media (48) for recording to form a recorded stream of data, the meta header data (82) including a cycle mark value marking cycle boundaries within the recorded stream of data.

The elements of Claim 44, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The method of writing data to a media storage device there comprises receiving a received packet (74) of data to be written to the media storage device (40), the received packet (74) of data including a packet header (76), adding a meta header (82) to the received packet (74) of data thereby forming an extended packet (80) of data which includes both the packet header (76) and the meta header (82), wherein the received packet (74)

of data is an isochronous packet of data received over an isochronous channel and storing the extended packet (80) of data onto a media (48) within the media storage device (40).

The elements of Claim 50, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The method of writing data to a media storage device described there comprises receiving a received packet (74) of data to be written to the media storage device (40), the received packet (74) of data including a packet header (76) and a common isochronous packet header (78), adding a meta data header (82) to the received packet (74) of data thereby forming an extended packet (80) of data which includes the packet header (76), the common isochronous packet header (78) and the meta data header (82) and storing the extended packet (80) of data onto a media (48) within the media storage device (40).

The elements of Claim 51, directed to one embodiment of the present invention, are described in the Specification at page 13, line 1 through page 15, line 10; page 16, line 17 through page 17, line 7; page 17, line 14 through page 18, line 14; page 20, line 24 through page 21, line 13; page 22, lines 1-19 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The media storage device described there comprises an interface circuit (42) configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, storage media (48) configured to store and retrieve the received stream of data, wherein the received stream of data includes one or more received packets (74) of data, each including both a packet header (76) and a common isochronous packet header (78) and an embedded stream processor (44) coupled to the interface circuit (42) and to the storage media (48) to add a meta data header (82) to each received packet (74) in the received stream of data as it is received, thereby forming an extended packet (80) of data, and provide the extended packet (80) of data to the storage media (48) for recording to form a recorded stream of data, the meta data header (82) including a cycle mark value marking cycle boundaries within the recorded stream of data.

The elements of Claim 52, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The method of writing data to a media storage device described there comprises receiving a received packet (74) of data to be written to the media storage device (40), the received packet (74) of data including a packet header (76), wherein the media storage device (40) maintains the packet header (76) with the received packet

(74) of data, adding a meta data header (82) to the received packet (74) of data thereby forming an extended packet (80) of data including both the packet header (76) and the meta data header (82) and storing the extended packet (80) of data onto a media (48) within the media storage device (40).

The elements of Claim 53, directed to one embodiment of the present invention, are described in the Specification at page 11, line 23 through page 12, line 10; page 13, lines 21-26; page 15, line 11 through page 16, line 26; page 20, line 24 through page 21, line 14 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The meta data header (82) added to received packets (74) by a media storage device (40) as the packets (74) are recorded on storage media (48) within the media storage device (40), each of the received packets (74) including an existing header (76), wherein the media storage device (40) maintains the existing header (76) with the received packets (74), the meta data header (82) described there comprises a cycle mark value including a pattern used to locate cycle boundaries within the received packets (74) and a cycle count value specifying a cycle number of a cycle in which the received packets (74) are received.

The elements of Claim 54, directed to one embodiment of the present invention, are described in the Specification at page 13, line 1 through page 15, line 10; page 16, line 17 through page 17, line 7; page 17, line 14 through page 18, line 14; page 20, line 24 through page 21, line 13; page 22, lines 1-19 and the accompanying Figures 3, 4A, 4B, 5, 7 and 10. The media storage device described there comprises an interface circuit (42) configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data (76), storage media (48) configured to store and retrieve the received stream of data and an embedded stream processor (44) coupled to the interface circuit (42) and to the storage media (40) to add meta header data (82) to the received stream of data as it is received and provide the meta header data (82) and the received stream of data, including the packet header data (76), to the storage media (40) for recording to form a recorded stream of data, the meta header data (82) including a cycle mark value marking cycle boundaries within the recorded stream of data.

VI. GROUNDS OF REJECTION AND OTHER MATTERS TO BE REVIEWED ON APPEAL

The following issues are presented in this Appeal Brief for review by the Board of Patent Appeals and Interferences:

1. Whether Claims 1, 24, 30, 44 and 50-54 are properly rejected under 35 U.S.C. § 102(e) as being anticipated by Kuver.
2. Whether Claims 1-13, 19-26, 29-32, 35 and 44-54 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Applicant Admitted Prior Art (AAPA) in view of Traw.
3. Whether Claims 14, 15, 27, 28, 33 and 34 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over AAPA in view of Traw and further in view of Kuver.

VII. ARGUMENT

Grounds for Rejection

Within the Office Action, Claims 1, 24, 30, 44 and 50-54 have been rejected under 35 U.S.C. § 102(e) as being anticipated by Kuver.

Outline of Arguments

In the discussion that follows, the Applicants discuss the teachings of Kuver. As will be discussed in detail below, Kuver does not teach that when receiving a packet, the packet header is maintained with the packet and a meta data header is added to the received packet.

1. Kuver does not teach when receiving a packet, the packet header is maintained with the packet and a meta data header is added to the received packet.

Kuver discloses a digital video network interface for transferring isochronous video data over an asynchronous local area network. It has become clear during the previous Office Actions and during the telephonic interview that the context of Kuver is not fully appreciated by the Examiner. This context of Kuver is very important to an appropriate understanding of what is being taught by Kuver. Kuver discusses a system, illustrated in Figure 2, that

> represents a system in which isochronous digital video data originating in a [sic] transmitting digital video (DV) camera is sent by a transmitting network interface across an asynchronous local area network, such as a Gigabit Ethernet network, to a receiving network interface which decodes and outputs the digital video data in isochronous mode to a receiving DV camera. [Kuver, col. 6, lines 47-53, Figure 2]

Kuver teaches a **transmitting-side** 1394 network interface 4, which is "a system for receiving digital video packets from the 1394 serial cable 2, *removing the 1394 data packet headers*, repackaging two or more (preferably three) data packets into network protocol format and transmitting the data over the asynchronous network 5." [Kuver, col. 7, lines 22-27, emphasis added] Kuver also teaches a **receiving-side** 1394 network interface which "receives and unpackages network data packets arriving from asynchronous network 5, reformats the video data into IEEE 1394 format and transfers the video data packets via the IEEE 1394 serial cable 7 to the isochronous receiving unit, DV camera 8." [Kuver, col. 7, lines 37-46] Kuver further teaches that

> [u]pon ***receiving*** the IEEE 1394 data packet, physical layer 15 transmits the data packet to link layer 16. Link layer 16 interprets the data in the data packet and removes all except data, meaning that *header information, header_CRC information and data_CRC information, **are all removed***. This leaves just the data field from the packet, which DMA 17 transmits to SDRAM 22. That is, link layer 16 interprets the IEEE 1394 header and the information regarding the data in the data packet in order to know where the data came from and where the data is going. **Link layer 16 then strips off unneeded information leaving only the data field.** [Kuver, col. 9, line 61 - col. 10, line 6, emphasis added]

Kuver therefore teaches that when ***receiving*** a packet, the header and all information other than the data is stripped from the packet. In contrast to the teachings of Kuver, as will be discussed in detail below, in the present invention, when ***receiving*** a packet, the packet header is maintained with the packet and a meta data header is added to the received packet.

Within the Final Office Action mailed April 14, 2006, the concepts of reception and transmission of packets are still confused. Within the rejections and the application of the cited references, there is still a fundamental confusion regarding the treatment of packets being received and the treatment of packets being transmitted. All of the citations provided from Kuver within the previous Office Actions are concerned with ***transmission*** of packets, not ***reception***. Specifically, in the cited section of Kuver at column 12, lines 52-59, ***transmission*** of

a packet from the interface out across the IEEE 1394 bus is being taught. In this cited section it is taught that

> DMA 17 accesses the SDRAM at the location of the empty pointer, and gives the digital video data to link layer 16 for reconstruction of the headers, shown in FIG. 3B. After reconstruction of the IEEE 1394 headers, link layer 16 gives the data, which is now formatted in accordance with IEEE 1394 protocol to physical layer 15 for ***isochronous transmission*** out across 1394 bus to a receiving digital video device. [Kuver, col. 12, lines 52-59, emphasis added]

In this section, Kuver is teaching adding an IEEE 1394 header to a packet of data and then **transmitting** the packet across the 1394 bus. This added IEEE 1394 header is the packet header necessary for ***transmission*** across the IEEE 1394 bus. Kuver does not teach adding a meta data header to a ***received*** packet of data which already included a packet header. Further, the cited Claim 14 of Kuver reads as follows

> 14. The method according to claim 13, wherein the step of converting isochronous digital video data includes ***removing IEEE 1394 header information*** from the digital video data, ***adding network header information*** to the digital video data, repackaging the digital video data with the network header into a network packet which is formatted in accordance with the local area network protocol format and storing the network packet for ***asynchronous output in the transmitting step***. [Kuver, col. 17, lines 38-46, emphasis added]

In this cited claim, Kuver teaches stripping the IEEE 1394 header information, adding network header information and then transmitting a packet over the network. Thus, Kuver teaches stripping the IEEE 1394 packet header and then adding the network header. Kuver does not teach adding a meta data header to a ***received*** packet, which already included a packet header. Again, this cited section of Kuver is being taken out of context. Claim 14 of Kuver is dependent on the independent Claim 5. The preamble of Claim 5 reads

> A method for transmitting digital video data, for use in a digital video conferencing system, in which digital video data is transmitted from one isochronous bus to another isochronous bus via an asynchronous bus, the method comprising the steps of: [Kuver, col. 16, lines 57-61]

As evident from the teachings of Kuver and this claim, the data is transmitted over several buses, each requiring an appropriate header. But, as taught within Kuver, and discussed in detail above,

at each receiving device between the buses, the received header is removed, the data is repackaged, and a header, appropriate for the next bus, is added to the data, before the packet is transmitted.

Within the Response to Arguments section of the Office Action mailed October 20, 2005, it was stated that "Kuver extends the packet of data by the added network header to the received digital video data (col. 17, line 41) and stores the extended packet of data in a transmitting step (col. 17, lines 45 and 46)." [Office Action mailed October 20, 2005, page 6] The applicants respectfully disagree. As discussed above, in column 17, lines 38-46 (Claim 14), Kuver does not teach extending a packet of data by adding a network header. Kuver first teaches "removing IEEE 1394 header information" and then "adding network header information." [Kuver, column 17, lines 38-46 (Claim 14)] **Both headers are not in the packet at the same time.** Therefore, Kuver does not teach extending the packet of data by adding a meta data header. Kuver teaches stripping one header, specific to one network (network the packet is received from), and then adding another header, specific to the other network (network the packet is transmitted on). Kuver does not teach adding a meta data header to a packet which still includes a packet header.

Furthermore, within the Response to Arguments section of the Office Action mailed October 20, 2005, it was stated that "Kuver does not need to remove all headers, only the unneeded headers are stripped (col. 12, lines 23-26)." The applicants respectfully disagree. The Examiner has quoted only part of the sentence. The entire sentence that the Examiner quoted in the Office Action mailed October 20, 2005 reads, "[T]hus, step S445 bypasses the protocol stack normally associated with network controller 23, and, after stripping off unneeded network headers and the like, transfers **only the data** directly to SDRAM 22." [Kuver, col. 12, lines 23-26, emphasis added] First of all, Kuver teaches stripping off unneeded network headers and the like, meaning more than just unneeded network headers are stripped off, otherwise the sentence would not include "and the like." Secondly, Kuver finishes the sentence with "transfers only the data directly to SDRAM 22." Only the data is transferred, nothing more. Hence no headers are transferred, so they must have been stripped off. The modifier "unneeded" is referring to all network headers, which are unneeded and is not referring to some network headers, as read by the Examiner. Otherwise, more than "only the data" would be transferred. This is clearly not what is taught by Kuver. The Applicants' interpretation of this sentence is also consistent with other parts of Kuver, such as the teachings that the "Link layer 16 interprets the data in the data packet and removes all except data, meaning that header information, header_CRC information

and data_CRC information, are all removed. This leaves just the data field from the packet, which DMA 17 transmits to SDRAM 22." [Kuver, col. 9, lines 62-67]

Within the Response to Arguments section of the Final Office Action, it is stated that Kuver discloses a received packet which includes a header and adding a network header information to the packet. It is also stated that Kuver extends the packet of data by the added network header to the receiver digital video data and stores the extended packet of data in a transmitting step. Within the Office Action it is also stated that Kuver further discloses formatting the digital video data with the header. [Office Action mailed April, 14, 2006, page 6] Each argument within the Final Office Action includes a cite to either Claims 14 or 15. As described above, the Examiner is leaving out crucial steps. Claim 14 discloses "adding network header information," but only after it also discloses "removing IEEE 1394 header information." [Kuver, Col, 17, lines 39-41 (Claim 14)] Claim 15 discloses "formatting the digital video data with IEEE 1394 header information," but only after it also discloses "removing the network header." [Kuver, Col, 17, lines 49-52 (Claim 15)] Therefore Kuver never teaches adding a meta header to the ***received*** packet of data thereby forming an extended packet of data including ***both*** the packet header and the meta data header.

Within the Response to Arguments section of the Final Office Action, the Examiner directs attention to Kuver's definition of a Header (col. 9, lines 39-41). [Office Action, mailed April 14, 2006, page 6] Again, directly following that, Kuver teaches "[u]pon removing the IEEE 1394 data packet, physical layer 15 transmits the data packet to link layer 16. Link layer 16 interprets the data in the data packet and removes all except data, meaning that header information, header_CRC information and data_CRC information are all removed. This just leaves the data field from the packet." [Kuver, col. 9, lines 61-66]

Within the Response to Arguments section of the Final Office Action, the Examiner also states that the packet header is reconstructed before transmission (col. 12, lines 52-59). [Office Action, mailed April 14, 2006, page 6] However, in the steps before that, specifically in the step S445, "unneeded network headers and the like" are stripped off and ***only*** the data is transfered. (emphasis added) [Kuver, col. 12, lines 23-26] Thus again, at no point does Kuver teach an extended packet of data including **both** the packet header and the meta data header. Furthermore, as previously described above and in other responses, the quoted language from Kuver relates to transmission not reception.

In contrast to the teachings of Kuver, the apparatus and method of the present invention ***receives*** a received packet of data to be written to the media storage device, adds a meta data header to the received packet of data thereby forming an extended packet of data, and stores the extended packet of data onto a media within the media storage device. **The extended packet of data includes both the packet header and the meta data header.** In one embodiment, referring to Figs. 4A and 4B, a series of source packets 60-63 is generated at a source device 50. The source device 50 then applies source packet headers 68-71 to each of the source packets 60-63, respectively. The source device 50 then splits the combination source packets and source packet headers into data blocks, with each source packet being split into multiple data blocks. Some number of the data blocks are then combined into an isochronous packet and the isochronous header and the common isochronous packet (CIP) header are then applied to the isochronous packet by the source device 50. Once the isochronous and CIP headers are applied to the isochronous data packet, the packet is then transmitted by the source device 50 over the IEEE 1394-1995 serial bus to the media storage device 40 of the present invention. **When the packet is received by the media storage device 40, a meta-data header is added by the media storage device 40 to the received packet**. As mentioned above, Kuver does not teach or disclose **adding a header** to the **received** packet of data. Kuver teaches stripping one header, specific to one network (network the packet is received from), and then adding another header, specific to the other network (network the packet is transmitted on).

2. The claims distinguish over Kuver.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claim 1

The independent Claim 1 is directed to a method of writing data to a media storage device. The method of Claim 1 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header, and storing the extended packet of data onto a media

within the media storage device. As described above, Kuver does not teach adding a meta header to the ***received*** packet of data thereby forming an extended packet of data including **both** the packet header and the meta data header. Kuver teaches stripping one header, specific to one network (network the packet is received from), and then adding another header, specific to the other network (network the packet is transmitted on). For at least these reasons, the independent Claim 1 is allowable over the teachings of Kuver.

b. Claim 24

The independent Claim 24 is directed to a media storage device. The media storage device of Claim 24 comprises means for interfacing configured for receiving a stream of data, thereby forming a received stream of data, and also for transmitting a retrieved stream of data, the received stream of data including packet header data, means for storing data for storing and retrieving the received stream of data, and means for processing coupled to the means for interfacing and to the means for storing for adding meta header data to the received stream of data as the received stream of data is received, such that each packet within the received stream of data includes both packet header data and meta header data, and providing the meta header data and the received stream of data to the means for storing for recording thereby forming a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, Kuver does not teach adding meta header data to the ***received*** stream of data, the received stream of data including packet header data, as the received stream of data is received and storing the header data and the received stream of data. For at least these reasons, the independent Claim 24 is allowable over the teachings of Kuver.

c. Claim 30

The independent Claim 30 is directed to a media storage device. The media storage device of Claim 30 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data, storage media configured to store and retrieve the received stream of data, and an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received, such that each packet within the received stream of data includes both packet header data and meta header data, and provide the meta header data and the received stream of data to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above,

Kuver does not teach an embedded stream processor to add meta header data to the ***received*** stream of data, which includes packet header data as it is received. Further, Kuver does not teach providing the meta header data and the received stream of data to the storage media for recording. For at least these reasons, the independent Claim 30 is allowable over the teachings of Kuver.

c. Claim 44

The independent Claim 44 is directed to a method of writing data to a media storage device. The method of Claim 44 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, adding a meta header to the received packet of data thereby forming an extended packet of data which includes both the packet header and the meta header, wherein the received packet of data is an isochronous packet of data received over an isochronous channel, and storing the extended packet of data onto a media within the media storage device. As described above, Kuver does not teach adding a meta header to the ***received*** packet of data, the received packet of data including a packet header, thereby forming an extended packet of data which includes both the packet header and the meta header and storing the extended packet of data onto a media within the media storage device. For at least these reasons, the independent Claim 44 is allowable over the teachings of Kuver.

d. Claim 50

The independent Claim 50 is directed to a method of writing data to a media storage device. The method of Claim 50 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header and a common isochronous packet header, adding a meta data header to the received packet of data thereby forming an extended packet of data which includes the packet header, the common isochronous packet header and the meta data header and storing the extended packet of data onto a media within the media storage device. As described above, Kuver does not teach adding a meta data header to a ***received*** packet that includes a packet header and a common isochronous packet header to form an extended packet of data and storing the extended packet of data onto a media within a media storage device. For at least these reasons, the independent Claim 50 is allowable over the teachings of Kuver.

e. Claim 51

The independent Claim 51 is directed to a media storage device. The media storage device of Claim 51 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, storage media configured to store and retrieve the received stream of data, wherein the received stream of data includes one or more received packets of data, each including both a packet header and a common isochronous packet header, and an embedded stream processor coupled to the interface circuit and to the storage media to add a meta data header to each received packet in the received stream of data as it is received, thereby forming an extended packet of data, and provide the extended packet of data to the storage media for recording to form a recorded stream of data, the meta data header including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, Kuver does not teach receiving a stream of data including one or more received packets, each including **both** a packet header and a common isochronous packet header and adding a meta data header to each ***received*** packet in the received stream of data. For at least these reasons, the independent Claim 51 is allowable over the teachings of Kuver.

f. Claim 52

The independent Claim 52 is directed to a method of writing data to a media storage device. The method of Claim 52 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, wherein the media storage device maintains the packet header with the received packet of data, adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header and storing the extended packet of data onto a media within the media storage device. As described above, Kuver does not teach adding a meta data header to the ***received*** packet of data thereby forming an extended packet of data including **both** the packet header and the meta data header. For at least these reasons, the independent Claim 52 is allowable over the teachings of Kuver.

g. Claim 53

The independent Claim 53 is directed to a meta data header added to received packets by a media storage device as the packets are recorded on storage media within the media storage device, each of the received packets including an existing header, wherein the media storage

device maintains the existing header with the received packets. The meta data header of Claim 53 comprises a cycle mark value including a pattern used to locate cycle boundaries within the received packets and a cycle count value specifying a cycle number of a cycle in which the received packets are received. As described above, Kuver does not teach a meta data header added to a ***received*** packet including an existing header. Kuver teaches stripping one header, specific to one network (network the packet is received from), and then adding another header, specific to the other network (network the packet is transmitted on). Further, Kuver does not teach a meta data header with a cycle mark value and a cycle count value. For at least these reasons, the independent Claim 53 is allowable over the teachings of Kuver.

h. Claim 54

The independent Claim 54 is directed to a media storage device. The media storage device of Claim 54 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data, storage media configured to store and retrieve the received stream of data and an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received and provide the meta header data and the received stream of data, including the packet header data, to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, Kuver does not teach receiving a stream of data including one or more received packets, each including **both** a packet header and a common isochronous packet header and adding a meta data header to each ***received*** packet in the received stream of data. For at least these reasons, the independent Claim 54 is allowable over the teachings of Kuver.

Grounds for Rejection

Within the Office Action, Claims 1-13, 19-26, 29-32, 35 and 44-54 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Applicant Admitted Prior Art (AAPA) in view of Traw.

Outline of Arguments

In the discussion that follows, the Applicants first discuss the teachings of AAPA, the teachings of Traw and the teachings of the combination of AAPA and Traw. As will be discussed in detail below, the combination of AAPA and Traw do not teach adding a meta data

header to the received packet of data which includes a packet header, thereby forming an extended packet of data which includes **both** the packet header and the meta data header.

1. AAPA does not teach adding a meta data header to the received packet of data which includes a packet header, thereby forming an extended packet of data which includes both the packet header and the meta data header.

Referring to Figure 4A of the present invention, which is designated as prior art, and the accompanying description, the isochronous and CIP headers are added to the isochronous data packet before the packet is ***transmitted*** by the source device 50 over the IEEE 1394-1995 serial bus to the media storage device 40 of the present invention. Referring now to Figure 4B, which is in accordance with the present invention, not the prior art, a new header (the meta data header) is added by the media storage device 40 after the packet is ***received*** by the media storage device thereby forming an extended packet, and storing the extended packet on a media storage device. The extended packet includes **both** the packet header and the added meta data header. AAPA does not teach or disclose adding a header after the packet is ***received*** by a media storage device thereby forming an extended packet of data which includes **both** the packet header and the added meta data header. Rather, AAPA simply teaches that the isochronous and CIP headers are inserted by the source device **prior to transmission** on the sending side. Specifically, AAPA provides, "[o]nce the isochronous and CIP headers are applied to the isochronous data packet, the packet is **then transmitted** by the source device 50 over the IEEE 1394-1995 serial bus to the media storage device 40." [present specification, page 13, lines 4-6, emphasis added] The corresponding figure, Figure 4A, includes adding headers; however, as quoted above, the headers are added before the packet is **transmitted**, not after it is received. Again, within the Final Office Action, the concepts of transmission and reception are being fundamentally confused and misapplied. For AAPA to teach the present invention, the AAPA would have to teach headers added to the isochronous data packet after the packet is received, not before the packet is transmitted. Hence, what Applicant admitted on page 12, lines 12 and 13, of the Preliminary Amendment filed August 6, 2004 (a copy of which is attached as Exhibit C) does not disclose the present invention. The language that the Examiner quotes regarding page 12, lines 12 and 13, of the Preliminary Amendment, highlights exactly the misunderstanding between transmission and reception discussed above. The language quoted in the Preliminary Amendment states, "CIP headers are added to the isochronous data packet **before** the packet is **transmitted**." Therefore, if the headers are added before transmission, they are not added after the packet is **received**. Hence, AAPA does not teach adding a meta data header to the **received** packet of data. The

language of the present specification used for AAPA is very explicit regarding this, it has been quoted above and should not be confused. Clearly, the headers in AAPA are added before the packet is transmitted, thus not added to a **received** packet of data.

Within the Final Office Action, the Examiner again appears to misunderstand the Present Invention. The Examiner first states, "Applicant admits that in AAPA, CIP headers are added to the isochronous data packet ***before*** the packet is transmitted." [Office Action mailed April 14, 2006, page 7] Then, the Examiner states, "[i]n summ[a]ry, AAPA adds metadata headers and stores packets with headers." [Office Action mailed April 14, 2006, page 7] There are many problems with this summary. First of all, AAPA teaches CIP headers, not meta data headers, so the summary somehow jumped to the incorrect conclusion that CIP headers are the same as meta data headers. Secondly, again, the CIP headers are added to the isochronous data packet ***before*** the packet is transmitted. The present invention is directed to adding a meta data packet ***after*** the packet is received. Thus, AAPA clearly does not teach adding a meta data header to the received packet of data which includes a packet header, thereby forming an extended packet of data which includes both the packet header and the meta data header.

2. Traw does not teach adding a meta data header to the received packet of data which includes a packet header, thereby forming an extended packet of data which includes both the packet header and the meta data header. Traw also does not teach storing the extended packet of data onto a media within the media storage device.

As recognized within the previous Office Actions, Traw also does not teach adding a header to a received packet of data thereby forming an extended packet of data. Traw also does not teach storing the extended packet of data onto a media within the media storage device.

3. The combination of AAPA and Traw does not teach adding a meta data header to the received packet of data which includes a packet header, thereby forming an extended packet of data which includes both the packet header and the meta data header.

In contrast to the teachings of the specification of the present invention which is designated as prior art, Traw and their combination, the method of and apparatus for writing and reading time sensitive data within a storage device of the present invention receives a received packet of data to be written to the media storage device, adds a header to the received packet of data thereby forming an extended packet of data, and stores the extended packet of data onto a media within the media storage device. **The extended packet of data includes both the packet header and the meta data header.** Referring to Figure 4A of the present application, the data

packet 80, prior to transmission by the source device 50, includes the Isoch header, the CIP header and the data blocks, as described previously. However, after ***receipt*** of the packet of data, the present invention adds the meta-data header 82 (Figure 4B), in contrast to the configuration described in either the specification of the present invention which is designated as prior art, Traw, or their combination. As described above, neither the specification of the present invention which is designated as prior art, Traw, nor their combination, teach receiving a received packet of data to be written to the media storage device, adding a header to the ***received*** packet of data thereby forming an extended packet of data and storing the extended packet of data onto a media within the media storage device.

As clearly described above using quoted language, AAPA only teaches adding headers to a packet of data **before** the packet is **transmitted**, thus it does not teach adding a header to a **received** packet of data. Hence, AAPA does not teach or make obvious the present invention. It is recognized within the Office Action that Traw does not teach adding a header to a received packet of data. Hence, Traw does not teach or make obvious the present invention. Accordingly, neither AAPA, Traw nor their combination teach or make obvious the present invention.

4. The claims distinguish over AAPA, Traw and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 1-7

The independent Claim 1 is directed to a method of writing data to a media storage device. The method of Claim 1 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header, and storing the extended packet of data onto a media within the media storage device. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach adding a meta data header to the ***received*** packet of data which includes a packet header, thereby forming an extended packet of data which includes **both** the packet header and the meta data header, and storing the extended packet of data onto a media within the media storage device. For at least these reasons, the independent Claim 1 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 2-7 are all dependent on the independent Claim 1. As discussed above, the independent Claim 1 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 2-7 are all also allowable as being dependent on an allowable base claim.

b. Claims 8-13

The independent Claim 8 is directed to a method of reading data from a media storage device which has previously been stored with header data generated by the media storage device. The method of claim 8 comprises locating a first header data, including a cycle mark value having a pattern, reading a previously stored packet of data following the first header data from a media within the media storage device, the previously stored packet of data including a packet header, stripping the first header data from the previously stored packet of data thereby forming a retrieved packet of data, and transmitting the retrieved packet of data to another device. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach generating header data by a media storage device, stripping the first header data from the previously stored packet of data which includes a packet header, thereby forming a retrieved packet of data and transmitting the retrieved packet of data to another device. For at least these reasons, the independent Claim 8 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 9-13 are all dependent on the independent Claim 8. As discussed above, the independent Claim 8 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 9-13 are all also allowable as being dependent on an allowable base claim.

c. Claims 19-23

The independent Claim 19 is directed to a meta data header added to received packets by a media storage device as the packets are recorded on storage media within the media storage device, each of the received packets including an existing header to which the meta data header is added such that the received packets include both an existing header and a meta data header. The meta data header of Claim 19 comprises a cycle mark value including a pattern used to locate cycle boundaries within the received packets and a cycle count value specifying a cycle number of a cycle in which the received packets are received. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their

combination teach adding a meta data header to an existing header of ***received*** packets by a media storage device, a cycle mark value including a pattern used to locate cycle boundaries within the received packets and a cycle count value specifying a cycle number of a cycle in which the received packets are received. For at least these reasons, the independent Claim 19 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 20-23 are all dependent on the independent Claim 19. As discussed above, the independent Claim 19 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 20-23 are all also allowable as being dependent on an allowable base claim.

d. Claims 24-26 and 29

The independent Claim 24 is directed to a media storage device. The media storage device of Claim 24 comprises means for interfacing configured for receiving a stream of data, thereby forming a received stream of data, and also for transmitting a retrieved stream of data, the received stream of data including packet header data, means for storing data for storing and retrieving the received stream of data, and means for processing coupled to the means for interfacing and to the means for storing for adding meta header data to the received stream of data as the received stream of data is received, such that each packet within the received stream of data includes both packet header data and meta header data, and providing the meta header data and the received stream of data to the means for storing for recording thereby forming a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach a means for processing for adding meta header data to the ***received*** stream of data which includes packet header data, as the received stream of data is received and providing the meta header data and the received stream of data to the means for storing for recording thereby forming a recorded stream of data. For at least these reasons, the independent Claim 24 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 25, 26 and 29 are all dependent on the independent Claim 24. As discussed above, the independent Claim 24 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 25, 26 and 29 are all also allowable as being dependent on an allowable base claim.

e. Claims 30-32 and 35

The independent Claim 30 is directed to a media storage device. The media storage device of Claim 30 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data, storage media configured to store and retrieve the received stream of data, and an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received, such that each packet within the received stream of data includes both packet header data and meta header data, and provide the meta header data and the received stream of data to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach an embedded stream processor to add meta header data to the ***received*** stream of data which includes packet header data, as it is received and provide the meta header data and the received stream of data to the storage media for recording to form a recorded stream of data. For at least these reasons, the independent Claim 30 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 31, 32 and 35 are all dependent on the independent Claim 30. As discussed above, the independent Claim 30 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 31, 32 and 35 are all also allowable as being dependent on an allowable base claim.

f. Claims 44-49

The independent Claim 44 is directed to a method of writing data to a media storage device. The method of Claim 44 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, adding a meta header to the received packet of data thereby forming an extended packet of data which includes both the packet header and the meta header, wherein the received packet of data is an isochronous packet of data received over an isochronous channel, and storing the extended packet of data onto a media within the media storage device. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach adding a meta header to the ***received*** packet of data which includes a packet

header, thereby forming an extended packet of data which includes **both** the packet header and the meta header and storing the extended packet of data onto a media within the media storage device. For at least these reasons, the independent Claim 44 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Claims 45-49 are all dependent on the independent Claim 44. As discussed above, the independent Claim 44 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 45-49 are all also allowable as being dependent on an allowable base claim.

g. Claim 50

The independent Claim 50 is directed to a method of writing data to a media storage device. The method of Claim 50 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header and a common isochronous packet header, adding a meta data header to the received packet of data thereby forming an extended packet of data which includes the packet header, the common isochronous packet header and the meta data header and storing the extended packet of data onto a media within the media storage device. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach adding a meta data header to a ***received*** packet that includes a packet header and a common isochronous packet header to form an extended packet of data which includes the packet header, the common isochronous packet header and the meta data header and storing the extended packet of data onto a media within a media storage device. For at least these reasons, the independent Claim 50 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

h. Claim 51

The independent Claim 51 is directed to a media storage device. The media storage device of Claim 51 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, storage media configured to store and retrieve the received stream of data, wherein the received stream of data includes one or more received packets of data, each including both a packet header and a common isochronous packet header, and an embedded stream processor coupled to the interface circuit and to the storage media to add a meta data header to each received packet in the received

stream of data as it is received, thereby forming an extended packet of data, and provide the extended packet of data to the storage media for recording to form a recorded stream of data, the meta data header including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach receiving a stream of data each including a packet header and adding a meta data header to each ***received*** packet in the received stream of data. For at least these reasons, the independent Claim 51 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

i. Claim 52

The independent Claim 52 is directed to a method of writing data to a media storage device. The method of Claim 52 comprises receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, wherein the media storage device maintains the packet header with the received packet of data, adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header and storing the extended packet of data onto a media within the media storage device. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach adding a meta data header to the ***received*** packet of data which includes a packet header, thereby forming an extended packet of data which includes both the packet header and the meta data header, and storing the extended packet of data onto a media within the media storage device. For at least these reasons, the independent Claim 52 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

j. Claim 53

The independent Claim 53 is directed to a meta data header added to received packets by a media storage device as the packets are recorded on storage media within the media storage device, each of the received packets including an existing header, wherein the media storage device maintains the existing header with the received packets. The meta data header of Claim 53 comprises a cycle mark value including a pattern used to locate cycle boundaries within the received packets and a cycle count value specifying a cycle number of a cycle in which the received packets are received. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach a meta data header

added to a ***received*** packet including an existing header. Further, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach a meta data header with a cycle mark value and a cycle count value. For at least these reasons, the independent Claim 53 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

k. Claim 54

The independent Claim 54 is directed to a media storage device. The media storage device of Claim 54 comprises an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data, storage media configured to store and retrieve the received stream of data and an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received and provide the meta header data and the received stream of data, including the packet header data, to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. As described above, neither the specification of the present invention which is designated as prior art, Traw nor their combination teach to add meta header data to the ***received*** stream of data as it is received and provide the meta header data and the received stream of data, including the packet header data, to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data. For at least these reasons, the independent Claim 54 is allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination.

Grounds for Rejection

Within the Final Office Action, Claims 14, 15, 27, 28, 33 and 34 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over AAPA in view of Traw and further in view of Kuver. Applicant respectfully disagrees.

Claims 14 and 15 are both dependent on the independent Claim 8. Claims 27 and 28 are both dependent on the independent Claim 24. Claims 33 and 34 are both dependent on the independent Claim 30. As discussed above, the independent Claims 8, 24 and 30 are all allowable over the teachings of the specification of the present invention which is designated as prior art, Traw and their combination. Accordingly, the dependent Claims 14, 15, 27, 28, 33 and 34 are all also allowable as being dependent on an allowable base claim.

5. CONCLUSION

For the above reasons, it is respectfully submitted that the Claims 1-15, 19-35 and 44-54 are allowable over the cited prior art references. Therefore, a favorable indication is respectfully requested.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: July 24, 2006

By: Jonathan O. Owens
Jonathan O. Owens
Reg. No.: 37,902
Attorneys for Applicants

VIII. CLAIMS APPENDIX

This appendix includes a list of the claims under appeal.

1. A method of writing data to a media storage device comprising:
   a. receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header;
   b. adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header; and
   c. storing the extended packet of data onto a media within the media storage device.

2. The method as claimed in claim 1 wherein the header includes a cycle mark value which includes a pattern used to locate cycle boundaries, and a cycle count value specifying a cycle number of a cycle in which the received packet of data was received.

3. The method as claimed in claim 1 wherein the received packet of data is an isochronous packet of data received over an isochronous channel.

4. The method as claimed in claim 1 wherein receiving the received packet of data includes receiving packets of data on multiple channels and further wherein adding a header to the received packet of data includes grouping packets received on multiple channels within a same isochronous cycle into a cycle group of packets and adding the header to the cycle group of packets.

5. The method as claimed in claim 1 wherein adding a header to the received packet of data is performed by an embedded stream processor within the media storage device.

6. The method as claimed in claim 1 wherein the received packet of data is received from a bus structure which complies with a version of an IEEE 1394 standard.

7. The method as claimed in claim 1 wherein the media storage device is a hard disk drive.

8. A method of reading data from a media storage device which has previously been stored with header data generated by the media storage device comprising:
   a. locating a first header data, including a cycle mark value having a pattern;

b. reading a previously stored packet of data following the first header data from a media within the media storage device, the previously stored packet of data including a packet header;

c. stripping the first header data from the previously stored packet of data thereby forming a retrieved packet of data; and

d. transmitting the retrieved packet of data to another device.

9. The method as claimed in claim 8 wherein transmitting includes transmitting the manipulated packet of data onto a bus structure which complies with a version of an IEEE 1394 standard.

10. The method as claimed in claim 8 wherein the pattern is used to locate cycle boundaries, and the first header data further includes a cycle count value specifying a cycle number of a cycle in which the previously stored packet of data was received.

11. The method as claimed in claim 8 wherein the retrieved packet is an isochronous packet of data and is transmitted over an isochronous channel.

12. The method as claimed in claim 8 wherein stripping the first header data from the previously stored packet of data is performed by an embedded stream processor within the media storage device.

13. The method as claimed in claim 8 wherein the media storage device is a hard disk drive.

14. The method as claimed in claim 8 wherein locating the first header data, including a cycle mark value having a pattern includes locating the pattern within the previously stored data, then determining if a cycle count value within the first header data is within an appropriate range, determining if an isochronous header follows the first header data and then determining a data length value.

15. The method as claimed in claim 14 wherein the appropriate range is any number including and between 0 and 7999.

16-18. (canceled)

19. A meta data header added to received packets by a media storage device as the packets are recorded on storage media within the media storage device, each of the received packets including an existing header to which the meta data header is added such that the received packets include both an existing header and a meta data header, the meta data header comprising:

a. a cycle mark value including a pattern used to locate cycle boundaries within the received packets; and

b. a cycle count value specifying a cycle number of a cycle in which the received packets are received.

20. The meta data header as claimed in claim 19 wherein the cycle count value has a range between and including 0 and 7999.

21. The meta data header as claimed in claim 19 wherein the received packets are isochronous data packets.

22. The meta data header as claimed in claim 19 wherein the meta data header is added to each received packet.

23. The meta data header as claimed in claim 19 wherein the meta data header is added to each group of received packets received during a same isochronous cycle.

24. A media storage device comprising:

a. means for interfacing configured for receiving a stream of data, thereby forming a received stream of data, and also for transmitting a retrieved stream of data, the received stream of data including packet header data;

b. means for storing data for storing and retrieving the received stream of data; and

c. means for processing coupled to the means for interfacing and to the means for storing for adding meta header data to the received stream of data as the received stream of data is received, such that each packet within the received stream of data includes both packet header data and meta header data, and providing the meta header data and the received stream of data to the means for storing for recording thereby forming a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data.

25. The media storage device as claimed in claim 24 wherein the means for processing is an embedded stream processor which also locates a first cycle mark value within the recorded

stream of data during a playback operation, reads packets within the recorded stream of data after the first cycle mark value, strips the header data from read packets within the recorded stream of data thereby forming retrieved packets of data and transmits the retrieved packets of data through the means for interfacing to a receiving device.

26. The media storage device as claimed in claim 25 wherein the receiving device is coupled to the means for interfacing by a bus structure which complies with a version of an IEEE 1394 standard.

27. The media storage device as claimed in claim 25 wherein the embedded stream processor locates the first cycle mark value by locating a pattern included within the cycle mark value, then determining if a cycle count value within the header data is within an appropriate range, determining if an isochronous header follows the header data and then determining a data length value.

28. The media storage device as claimed in claim 27 wherein the appropriate range is any number including and between 0 and 7999.

29. The media storage device as claimed in claim 24 wherein the header data further includes a cycle count value specifying a cycle number of a cycle in which packets of data within the received stream of data were received.

30. A media storage device comprising:

a. an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data;

b. storage media configured to store and retrieve the received stream of data; and

c. an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received, such that each packet within the received stream of data includes both packet header data and meta header data, and provide the meta header data and the received stream of data to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data.

31. The media storage device as claimed in claim 30 wherein the embedded stream processor also locates a first cycle mark value within the recorded stream of data during a playback operation, reads packets within the recorded stream of data after the first cycle mark value, strips the header data from read packets within the recorded stream of data thereby forming retrieved packets of data and transmits the retrieved packets of data through the interface circuit to a receiving device.

32. The media storage device as claimed in claim 31 wherein the receiving device is coupled to the media storage device by a bus structure which complies with a version of an IEEE 1394 standard.

33. The media storage device as claimed in claim 31 wherein the embedded stream processor locates the first cycle mark value by locating a pattern included within the cycle mark value, then determining if a cycle count value within the header data is within an appropriate range, determining if an isochronous header follows the header data and then determining a data length value.

34. The media storage device as claimed in claim 33 wherein the appropriate range is any number including and between 0 and 7999.

35. The media storage device as claimed in claim 30 wherein the header data further includes a cycle count value specifying a cycle number of a cycle in which packets of data within the received stream of data were received.

36-43. (canceled).

44. A method of writing data to a media storage device comprising:

a. receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header;

b. adding a meta header to the received packet of data thereby forming an extended packet of data which includes both the packet header and the meta header, wherein the received packet of data is an isochronous packet of data received over an isochronous channel; and

c. storing the extended packet of data onto a media within the media storage device.

45. The method as claimed in claim 44 wherein the header includes a cycle mark value which includes a pattern used to locate cycle boundaries, and a cycle count value specifying a cycle number of a cycle in which the received packet of data was received.

46. The method as claimed in claim 44 wherein receiving the received packet of data includes receiving packets of data on multiple channels and further wherein adding a header to the received packet of data includes grouping packets received on multiple channels within a same isochronous cycle into a cycle group of packets and adding the header to the cycle group of packets.

47. The method as claimed in claim 44 wherein adding a header to the received packet of data is performed by an embedded stream processor within the media storage device.

48. The method as claimed in claim 44 wherein the received packet of data is received from a bus structure which complies with a version of an IEEE 1394 standard.

49. The method as claimed in claim 44 wherein the media storage device is a hard disk drive.

50. A method of writing data to a media storage device comprising:

a. receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header and a common isochronous packet header;
b. adding a meta data header to the received packet of data thereby forming an extended packet of data which includes the packet header, the common isochronous packet header and the meta data header; and
c. storing the extended packet of data onto a media within the media storage device.

51. A media storage device comprising:

a. an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data;
b. storage media configured to store and retrieve the received stream of data, wherein the received stream of data includes one or more received packets of data, each including both a packet header and a common isochronous packet header; and
c. an embedded stream processor coupled to the interface circuit and to the storage media to add a meta data header to each received packet in the received stream of

data as it is received, thereby forming an extended packet of data, and provide the extended packet of data to the storage media for recording to form a recorded stream of data, the meta data header including a cycle mark value marking cycle boundaries within the recorded stream of data.

52. A method of writing data to a media storage device comprising:

a. receiving a received packet of data to be written to the media storage device, the received packet of data including a packet header, wherein the media storage device maintains the packet header with the received packet of data;

b. adding a meta data header to the received packet of data thereby forming an extended packet of data including both the packet header and the meta data header; and

c. storing the extended packet of data onto a media within the media storage device.

53. A meta data header added to received packets by a media storage device as the packets are recorded on storage media within the media storage device, each of the received packets including an existing header, wherein the media storage device maintains the existing header with the received packets, the meta data header comprising:

a. a cycle mark value including a pattern used to locate cycle boundaries within the received packets; and

b. a cycle count value specifying a cycle number of a cycle in which the received packets are received.

54. A media storage device comprising:

a. an interface circuit configured to receive a stream of data, thereby forming a received stream of data, and also to transmit a retrieved stream of data, the received stream of data including packet header data;

b. storage media configured to store and retrieve the received stream of data; and

c. an embedded stream processor coupled to the interface circuit and to the storage media to add meta header data to the received stream of data as it is received and provide the meta header data and the received stream of data, including the packet header data, to the storage media for recording to form a recorded stream of data, the meta header data including a cycle mark value marking cycle boundaries within the recorded stream of data.

IX. EVIDENCE APPENDIX

STATEMENT

Pursuant to 37 C.F.R. § 41.37(c)(1)(ix), the following is a statement setting forth where in the record the evidence of this appendix was entered by the examiner:

Evidence Description:	**Where Entered:**
U.S. Pat. No. 6,438,604	Office Action mailed April 1, 2003
U.S. Pat. No. 6,012,117	Office Action mailed April 1, 2003
Preliminary Amendment filed August 6, 2004	Office Action mailed November 2, 2004
Office Action October 20, 2005	Examiner Office Action
Office Action April 14, 2006	Examiner Office Action

X. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.